

Supplementary Figure 1: Stimulus information

- (a)** Examples of stimuli used to test face selectivity (Fig. 1a, b).
(b) (top row) 8 pictures of real faces; (middle row) corresponding brightness edge maps (obtained using the Prewitt method with a threshold of 0.05); (bottom row) corresponding cartoon faces constructed using 7 elementary features: hair outline, eyes, irises, eyebrows, nose, mouth, and face outline. Note how the edges define these elementary features. These specific cartoons were used in the experiments shown in Fig. 1c, and in a psychophysical experiment to assess whether such simple cartoons can capture face identity (part c below).
(c) Identification performance for cartoon faces, averaged across 8 subjects. Subjects were asked to match 16 real faces to 16 cartoon versions. Horizontal black line indicates chance performance (0.06). Each cartoon face shown in part b was correctly matched to its real counterpart by at least half of the subjects.
(d) Full set of 2^7 partial faces constructed from a set of 7 elementary features, used to analyze neuronal mechanisms for face detection (Fig. 2).

Supplementary Figure 2: Example of a time-resolved tuning profile

Example time resolved tuning curve of a cell significantly tuned to iris size. The time resolved tuning curve (centre) plots firing rate as a function of feature value and time after feature value update (post stimulus time). The firing rate (left, blue) was mildly modulated by the feature update cycle. But response heterogeneity (left, red) was strongly influenced by the feature values. Heterogeneity surpassed the significance threshold (left, dotted red line, see Methods) between 136 and 296 ms after feature update, peaking at 195 ms. The tuning curve at this delay (bottom) was modulated by more than a factor of two between the extreme feature values. For comparison, the average shuffle predictor tuning profile is shown on the right. It shows the same firing rate modulation after feature update as the original tuning profile, but no dependence on feature value.

Supplementary Figure 3: Comparisons of response and feature tuning time courses

Comparison of time courses of middle face patch neuron responses to **(a)** pictures of faces, cartoons and gadgets presented for 200 ms and with 200 ms ISI and **(b)** to RSVP cartoon face stimulus presentation with 116 ms presentation time. Population PSTH (a, left) and face selectivity index (FSI) time course (a, right) are causally smoothed with a half Gaussian of width 10ms. Responses to faces, cartoons and non-face objects start to differ about 80 ms after stimulus onset (a, left). At that point the face selectivity index rises steeply (a, right) and remains high during the entire response period. Peak face selectivity is reached around 170 ms post stimulus time. The population PSTH to RSVP stimulus presentation (b, upper left) is not much modulated. This is because different cells exhibit response maxima over a large range of post-stimulus times (b, lower left, which shows normalized PSTHs of all cells, sorted by the time delay of the maximal response). Significant *tuning* to cartoon features starts 75 ms after feature update and peaks 180 ms after feature update (b, upper right, heterogeneity of all significant tuning curves in blue, red lines mark time point of

emerging tuning (half maximal heterogeneity value); lower right: time courses of tuning, sorted by onset). Thus the time course of tuning to cartoons is comparable to the time course of development of face selectivity to real-world stimuli, despite the very different stimulus pacing in the two experiments. (a, same cells as in Fig. 1, b, same cells as in Figs. 3 and 4)

Supplementary Figure 4: Influence of fixation position on feature tuning

See Supplementary Text 1 for details.

Supplementary Figure 5: Relationship between incidence of tuning and physical size of feature changes

See Supplementary Text 2 for details.

Supplementary Figure 6: Tuning in neighboring cells

See Supplementary Text 3 for details.

Supplementary Figure 7: Singular value decomposition analysis and polynomial fitting of joint tuning functions

See Supplementary Text 4 for details.

Supplementary Figure 8: Cross-feature correlations

Matrix of all joint tuning curves of the example cell from Fig. 5a.

Supplementary Figure 9: Fitting a Gamma function to the shuffle heterogeneity distribution

Distributions of 5016 heterogeneity values obtained by reshuffling (at the relevant response delay) for four randomly chosen cells. Histograms are normalized to an area of 5016. The distributions are well fit by a gamma function (red curves). Using gamma curve fits, areal percentiles can be computed and significance thresholds (for $p=0.001$, i.e. 0.1% of the area) computed (light red arrow). This method does not require many samples. When we repeated the procedure for 16 subsets of the 5016 data points (each with 313 or 314 data points), we obtained 16 estimates for the significance threshold (marked by the small light red bars). While these values differ, their distribution is actually quite compact, demonstrating robustness. The significance threshold obtained by the method used in this paper is shown by the dark red arrow. In brief, this method is based on the five largest heterogeneity values of the reshuffle distribution. Only the ranking of values is considered and no assumption needs to be made about the shape of the distribution of the resampled heterogeneity values. In all four cells, this significance threshold is higher than that of the gamma fit based method. Thus, this method ensures that the actual heterogeneity value deemed significant, is at least on a par with the five largest values of the 5016 reshuffle samples, and it is robustly higher than the method based on the gamma distribution.

Supplementary Figure 10: Tuning in three example cells assessed by Gaussian fitting and Entropy-based methods.

See Supplementary Text 5 for details.

Supplementary Figure 11: Incidences of tuning as assessed by the Gaussian and Entropy-based methods.

See Supplementary Text 5 for details.

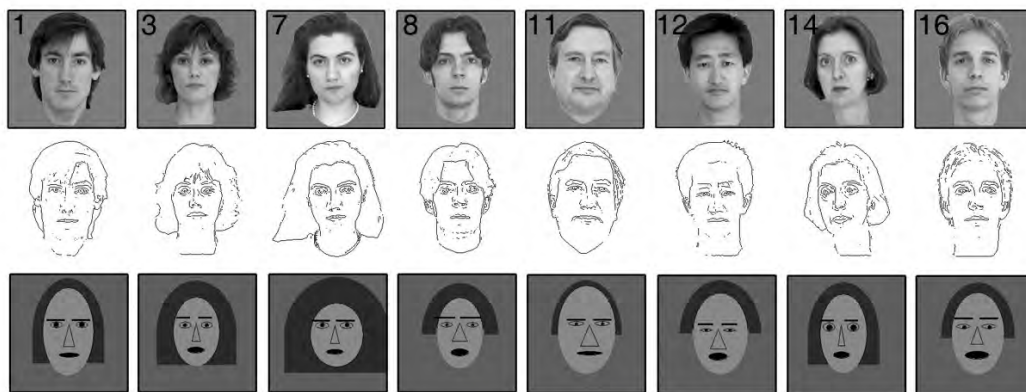
Supplementary Figure 12: Results of Gaussian fitting method

See Supplementary Text 5 for details.

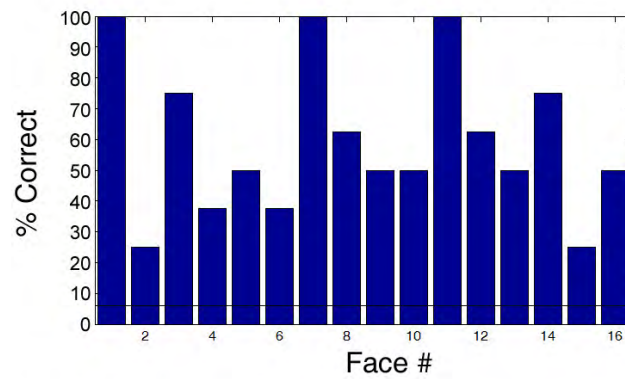
a



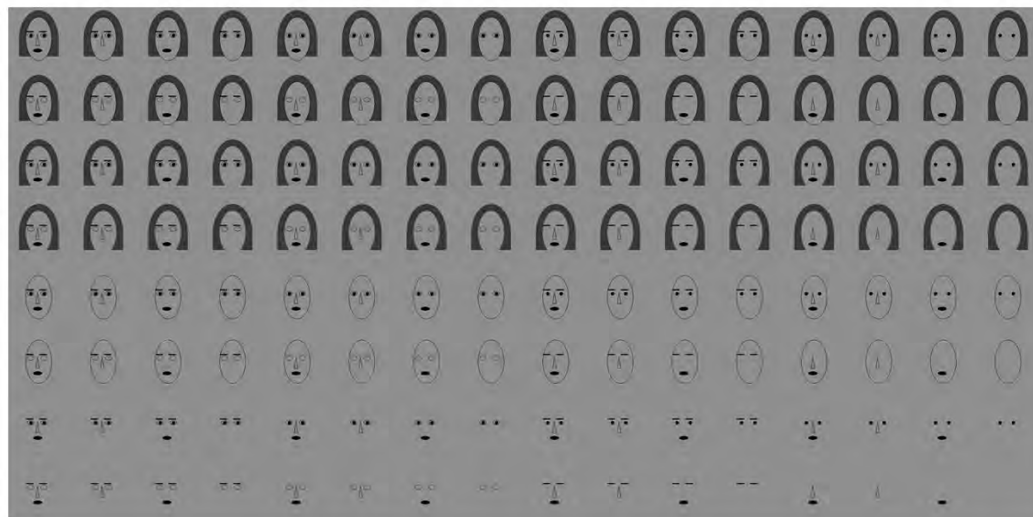
b

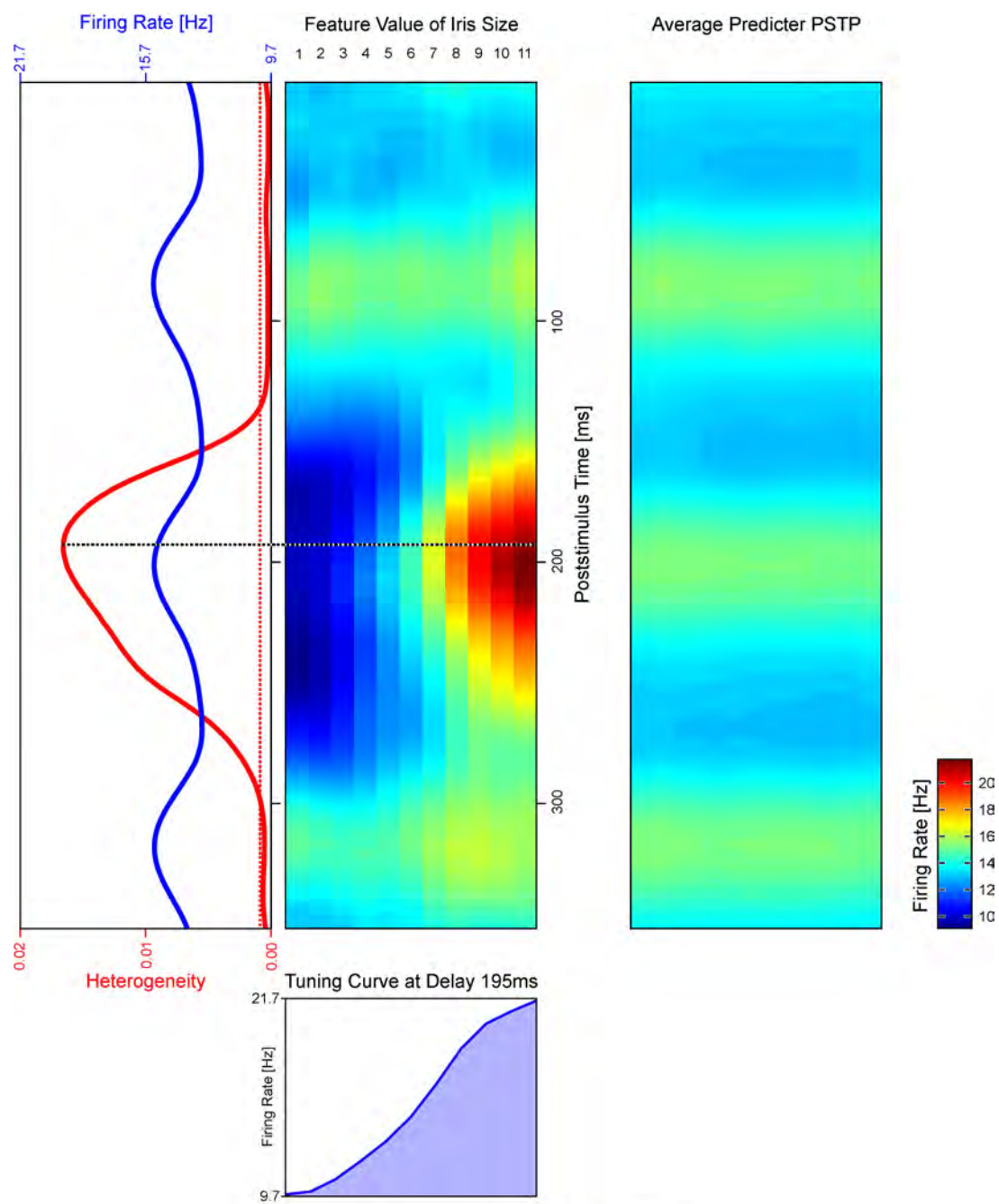


c



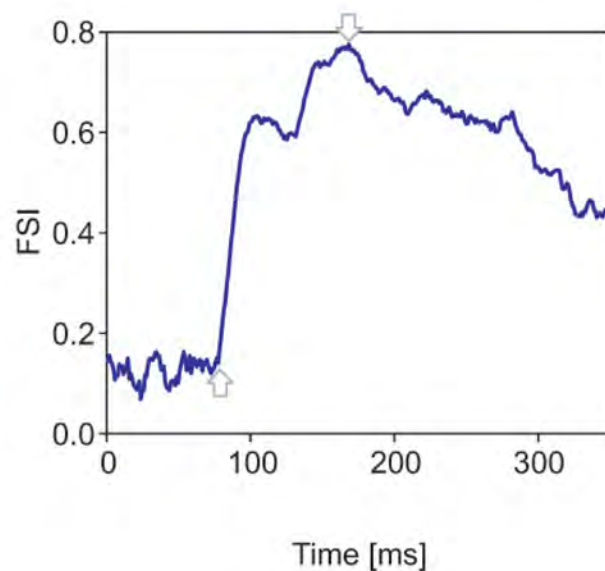
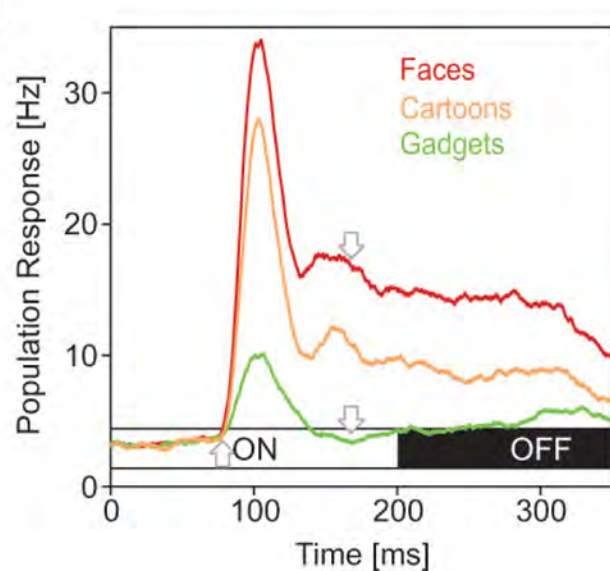
d



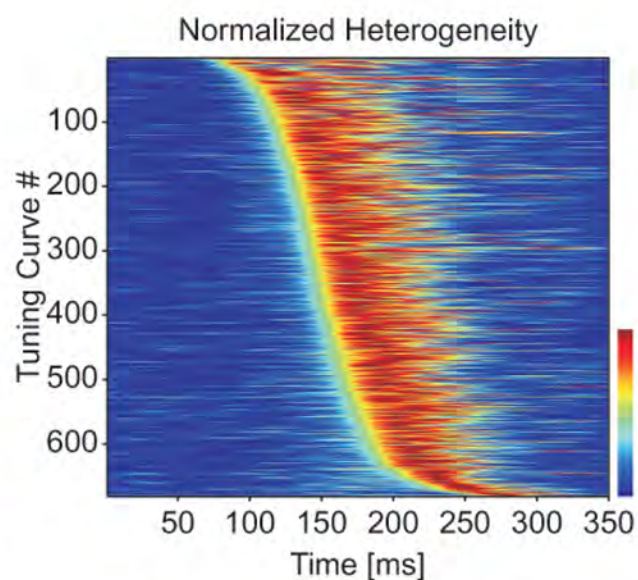
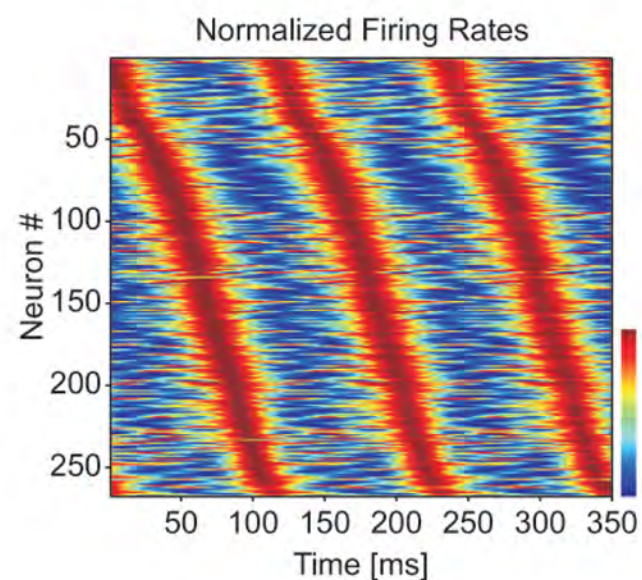
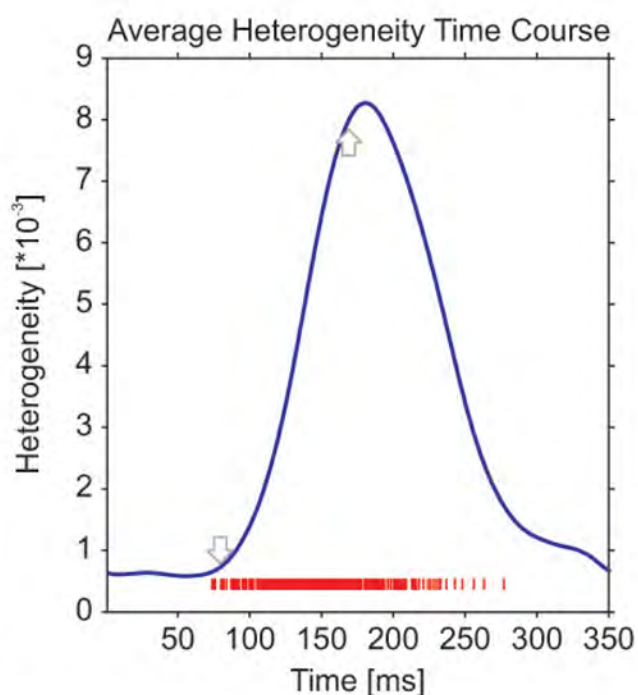
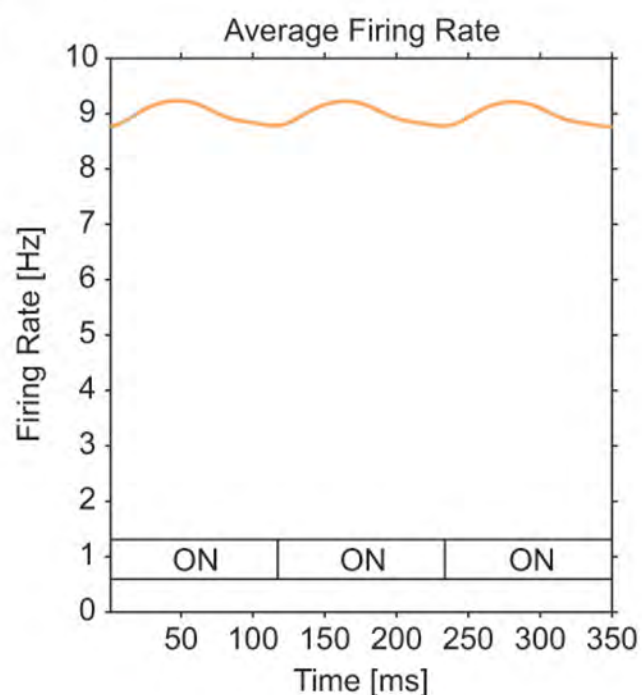


Supplementary Figure 2

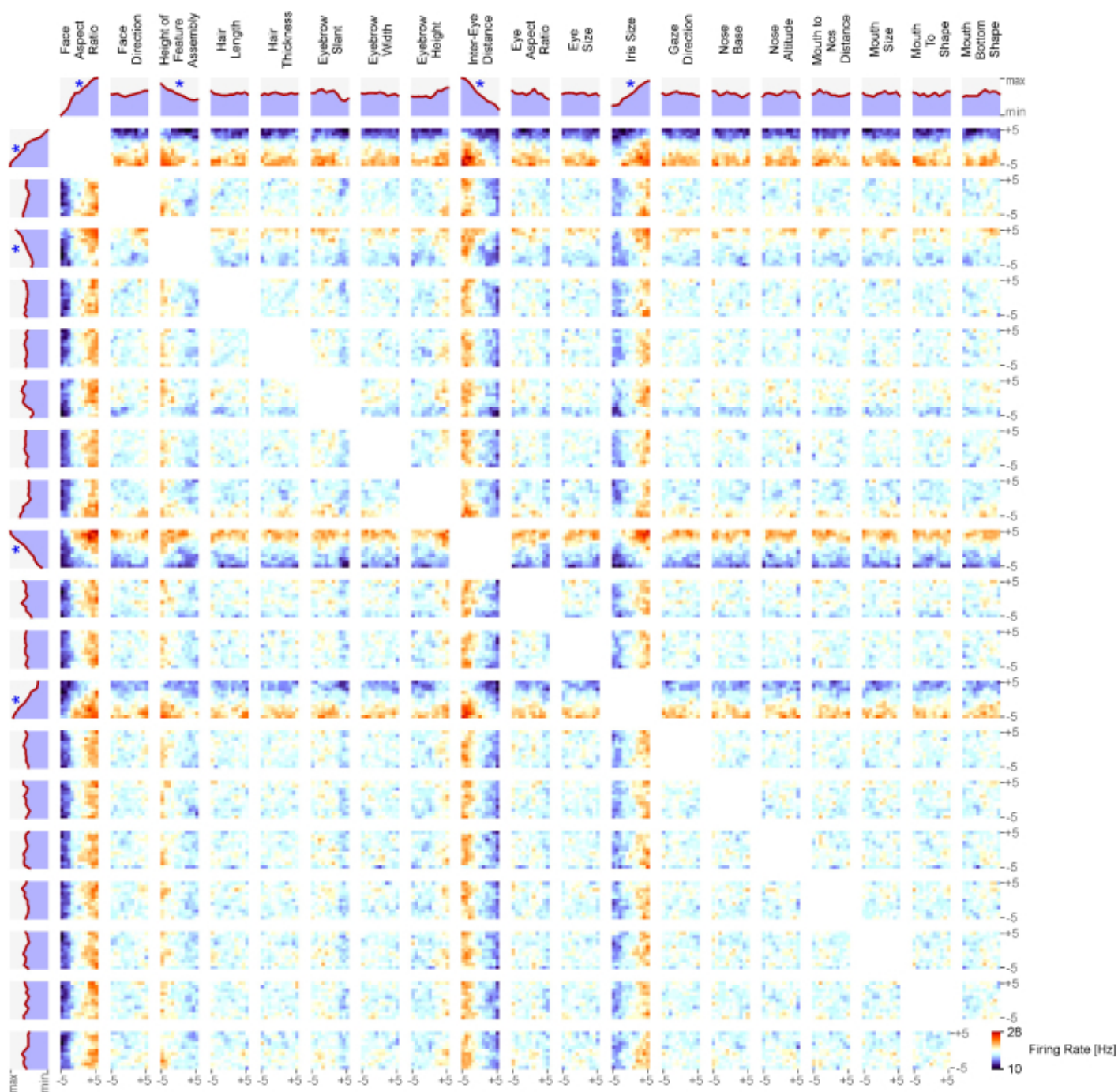
a



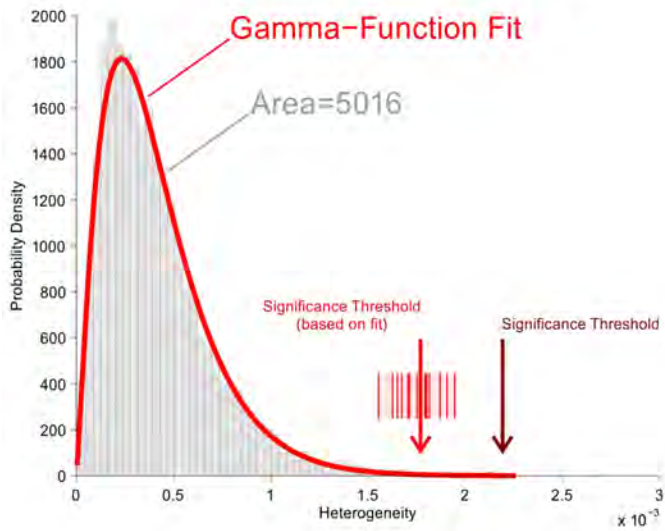
b



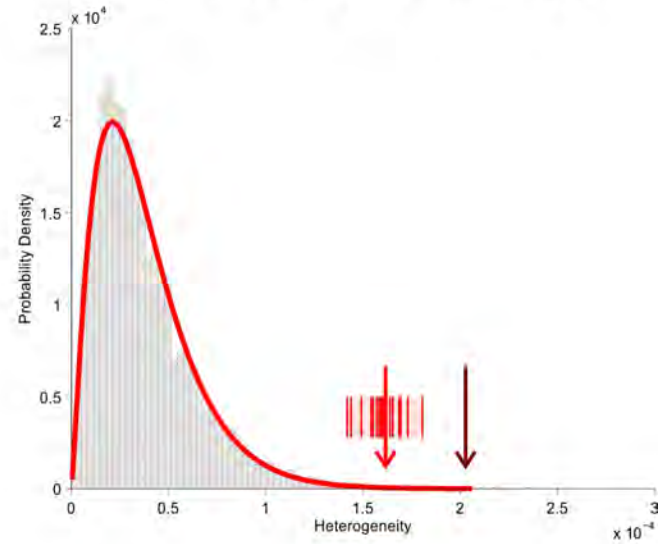
Supplementary Figure 3



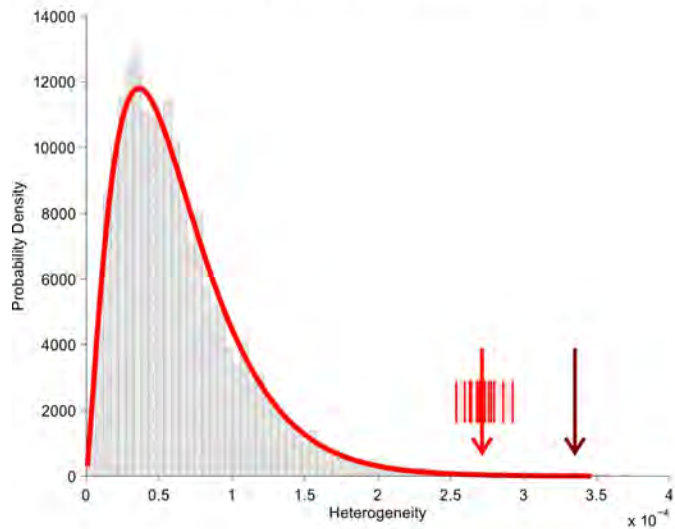
Heterogeneity Distribution Cell 1



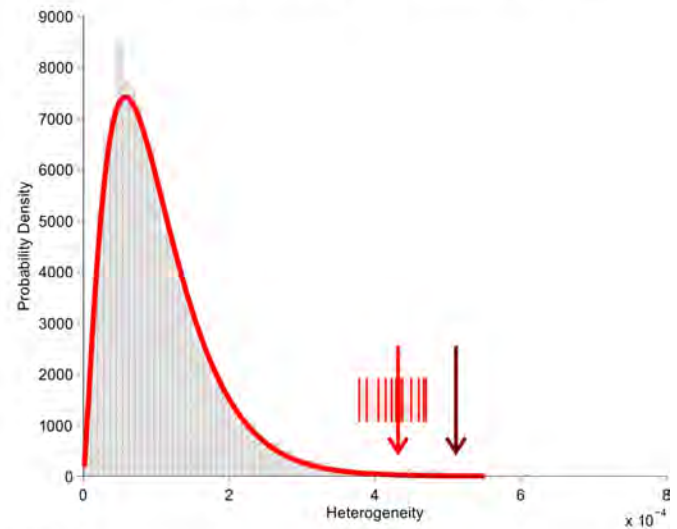
Heterogeneity Distribution Cell 2



Heterogeneity Distribution Cell 3



Heterogeneity Distribution Cell 4



Supplementary Figure 9

Supplementary Text 1: Tuning and Eye Positions

The paucity of tuning to nose- and mouth-related features compared to eye and eyebrow related ones, causes the concern that this may be the result of preferential looking to these features, possibly as a consequence of more attention being paid to these parameters than to others. To address this concern, we analyzed the dependence of tuning on eye positions.

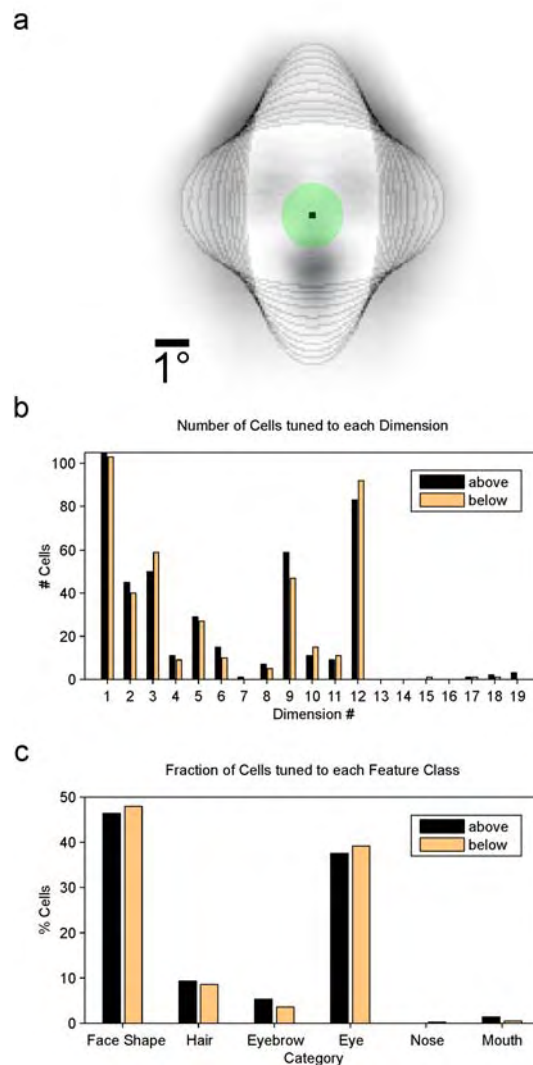
Monkeys foveated the central fixation spot with some horizontal and vertical deviation (horizontal mean cross-experiment standard deviation 0.78° , 0.92° , and 0.63° visual angle in monkeys A, T, and L, and vertical mean cross-experiment standard deviation of 0.93° , 0.97° , and 0.92° , respectively). Comparison of these figures to the average cartoon face (**Supplementary Fig. 4a**) shows that these deviations, though less than one degree of visual angle in all animals, could have brought the fovea closer to eyes and mouth and may have induced some bias for one parameter or the other. (Note that the prevalence of tuning for facial layout parameters and the higher incidence of tuning for hair than mouth or nose parameters cannot be explained this way.)

We then compared tuning during times when the monkey was foveating above the fixation spot, which should favor tuning to eye parameters, with tuning during times when he was foveating below, which should favor tuning to mouth parameters.

Incidences of tuning to features (**Supplementary Fig. 4b**) and categories (**Supplementary Fig. 4c**) were very similar in the two conditions (Pearson correlation coefficient of the two distributions in Fig.1b: $r=0.99$, $p<<0.001$). (There was slightly less tuning to eye related parameters during periods when the monkey's gaze was above fixation, and slightly less to mouth related parameters during fixations below, the opposite of what a gaze-direction explanation would predict.) This close match was found for each of the three monkeys (Pearson correlation coefficients of $r=0.98$, 0.97 , and

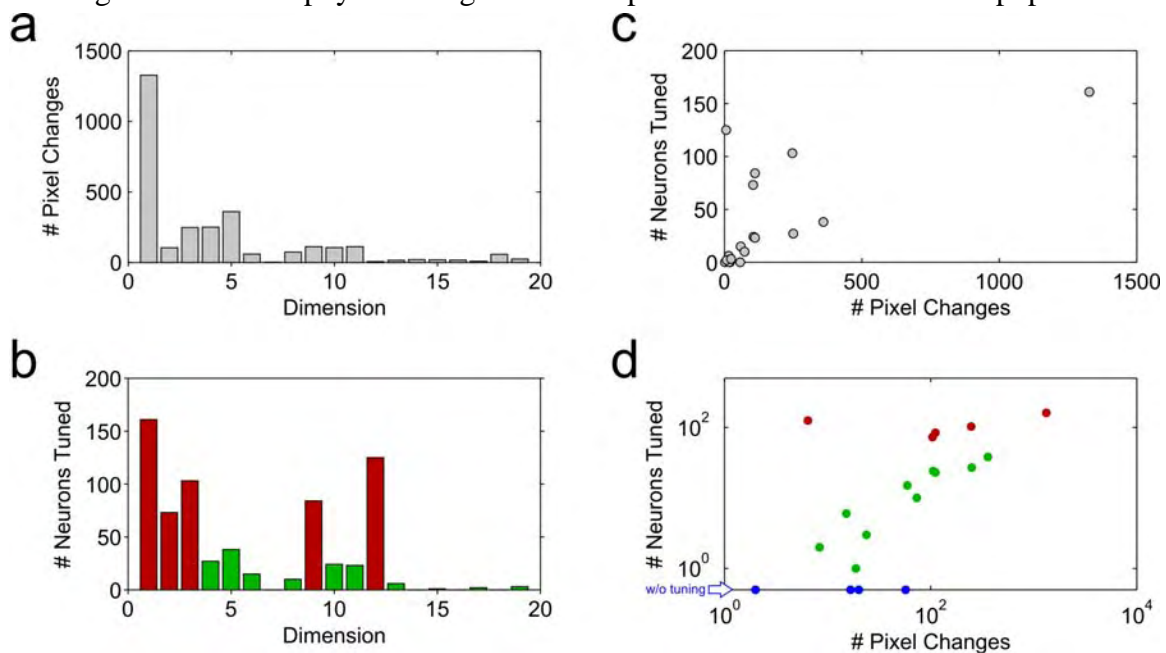
0.99, respectively, all highly significant $p < 0.001$), and in none any systematic or significant difference in tuning incidence to any of the features or categories was observed. Thus, there is no indication that eye position plays a role in determining the incidence of tuning to facial features in our experiment. Specifically, the hypothesis that preferential fixation on eyes may have caused more tuning to this parameter, cannot be substantiated.

Supplementary Figure 4 *Influence of fixation position on feature tuning. (a) Average cartoon face (mean across 500 random instantiations). Green circle indicates range of fixation positions (monkey T, one standard deviation). Incidence of tuning to the 19 feature dimensions (b) and six feature categories (c) during fixation above (black)*



Supplementary Text 2: Incidence of Tuning and Physical Size of Feature Changes

It is plausible that larger stimulus changes are more likely to induce tuning than smaller stimulus changes. We therefore analyzed, for each of the 19 different dimensions, the average number of pixel changes associated with any change along that feature dimension. As shown in **Supplementary Fig. 5a**, pixel changes were indeed quite different for the 19 different dimensions. This is expected, since the range of feature changes was set for each feature to span the entire range of physically possible faces, which is very different, of course, for, e.g., facial layout and nose size. The range of pixel values for face aspect ratio is the largest, those for iris size amongst the smallest. Despite this huge difference in physical range these two parameters are the two most popular ones



Supplementary Figure 5 Relationship between incidence of tuning and physical size of feature changes. (a) The average number of pixels differing between successive displays for each of the 19 dimensions. Step size was, by far, largest for face aspect ratio (dimension 1, conventions as in Figure 3d). The second biggest change occurs for change of hair thickness (dimension 5). (b) Number of neurons tuned to each of the 19 feature dimensions, the five most popular dimensions colored in red, the less popular dimensions in green. Iris size (dimension 12) is the second most popular dimension, yet, the physical change associated with it is the second smallest. (c) Scatter plot of neural tuning vs. pixel change for the 19 dimensions (a and b are the marginals). (d) Same scatter plot log scaled, the five most popular dimensions in red, less popular ones in green, and dimensions without tuning shown in blue. Pearson correlation coefficient is 0.67 for all 19 dimensions, and 0.92 for the ten intermediately popular dimensions (both highly significant $p < 0.001$).

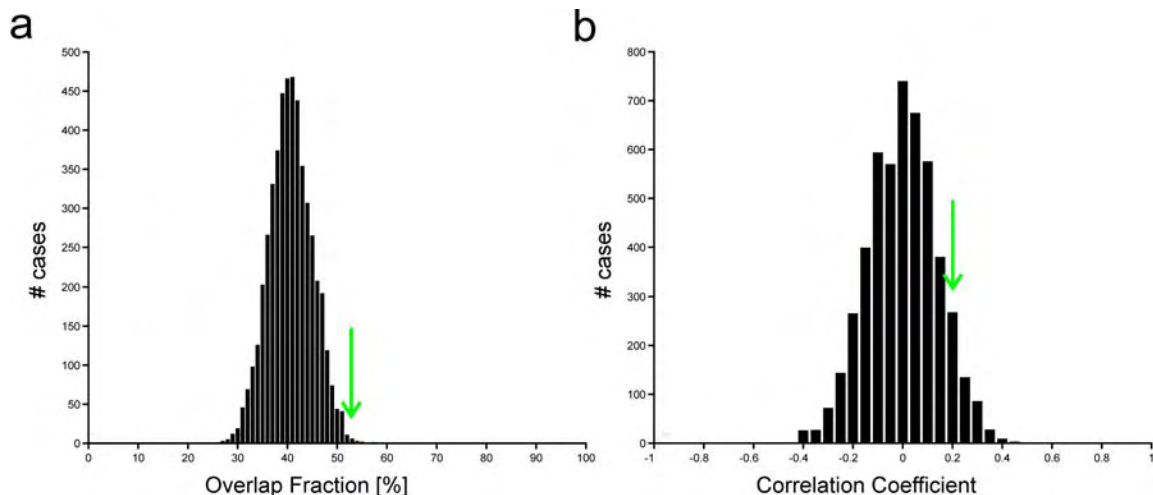
among middle face patch neurons (**Supplementary Fig. 5b**). Thus, the physical size of changes cannot explain all of the variance in tuning preference across dimensions. However, incidence of tuning is correlated with magnitude of physical change (**Supplementary Fig. 5c, d**). The Pearson correlation coefficient between tuning incidence and pixel change for the 19 dimensions is 0.67, i.e. 45% of the variance of tuning incidence can be explained by the range of physical feature variation. This correlation is strongest for the 10 parameters with relatively small number of neurons tuned ($r=0.92$, green bars in b and green dots in d), while for features without tuning (blue), trivially, and for features with a lot of tuning (red), no significant correlation was found.

Supplementary Text 3: Tuning in neighboring cells

We recorded from 52 pairs of neighboring cells. When one cell of a simultaneously recorded cell pair was significantly tuned to a set of feature dimensions, its neighbor was, on average, tuned to 53% of these dimensions as well. (For example, if cell 1 was tuned to dimensions 1, 5, and 7 and cell 2 was tuned to dimensions 5, 7 and 12, then the overlap fraction would be 67%.) This fraction was significantly larger than that for non-neighboring cell pairs (40%, $p < 0.01$, shuffle predictor); the distribution of 5000 shuffle predictors derived from random assignments of significantly tuned feature dimensions to the cells is shown in **Supplementary Fig. 6a** (green arrow marks experimentally determined fraction).

However, the total number of significantly tuned dimensions was only weakly correlated in neighboring cells ($r = 0.20$), overlapping with the distribution of shuffle predictors and not significant at $p = 0.05$ (**Supplementary Fig. 6b**).

Thus, there is some evidence for local clustering of shape selectivity within the middle face patches, but also of substantial tuning differences between neighbouring cells.



Supplementary Figure 6 Tuning in neighboring cells. The degree of overlap in feature tuning between neighboring cells (green arrows) was quantified in two ways and tested against 5000 shuffle distributions derived from random assignments of significantly tuned feature dimensions to 52 cells. (a) Fraction of dimensions two neighboring cells are both tuned to. (b) Correlation of the total numbers of significantly tuned dimensions in neighboring cells.

Supplementary Text 4: Singular value decomposition analysis and polynomial fitting of joint tuning functions

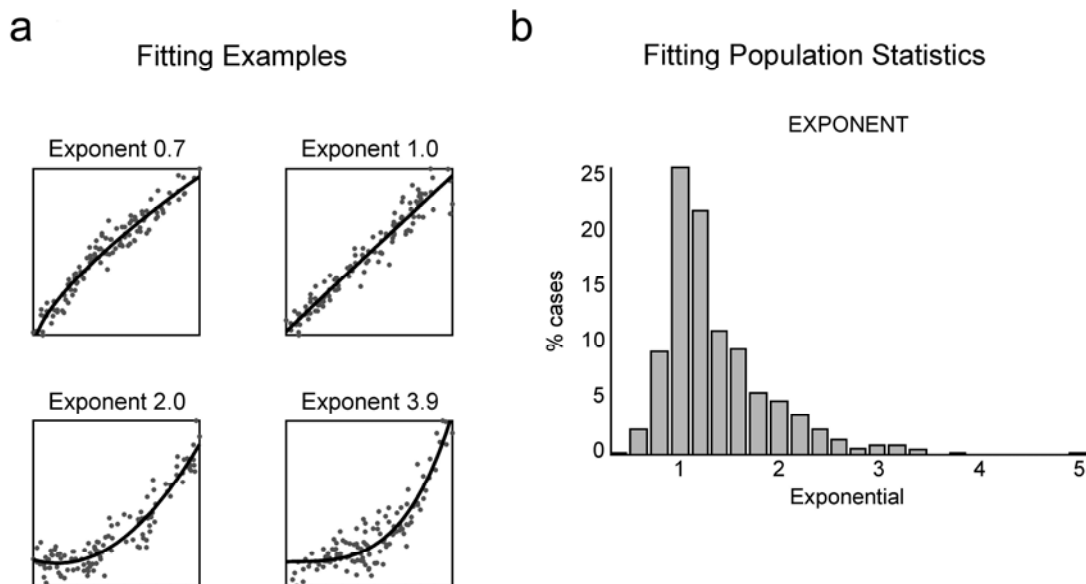
We found that the average middle face patch cell is tuned to 2.8 dimensions. How is tuning to these different dimensions integrated within each cell? As discussed in the paper, if feature integration is to preserve faithful measurement of individual features, then tuning to feature combinations should be separable into tuning to individual features¹.

To test for separability, we performed a singular value decomposition (SVD) of the joint tuning functions¹⁻³. SVD decomposes a joint tuning matrix into a sum of component matrices, each of which is the outer product of two single dimension functions. The weight of each matrix is called its singular value (SV). If this decomposition is dominated by one component, the joint tuning matrix is separable; if more than one component is necessary, then it is inseparable. We therefore tested for the strongest SV to be significant (using reshuffling procedures, $p < 0.01$) and the second strongest component to be insignificant ($p > 0.05$)¹. Using this method, we found all 771 joint tuning curves to be separable.

In the section “Face Differentiation: Integration of Features”, we show that integration schemes used by cells could be well modelled by both multiplication and addition of individual tuning curves; the average correlation coefficient between actual joint tuning functions and multiplicative predictors was 0.89, while for additive predictors, it was 0.88 (significantly lower ($p < 0.01$) than multiplicative predictors, Wilcoxon rank sum test). This means the integration scheme for each of the tuning curve pairs cannot be exactly determined. However, since we studied a large population of cells with 771 feature pairs, we can make some statements about which kind of separability

scheme the population shows. Even though for any given pair of dimensions, a certain integration scheme, say multiplicative, may not be distinguishable from another, say additive, with high confidence, if in the population one scheme predominates, this allows us to be confident about integration schemes at the population level.

In order to quantify the integration scheme used by each neuron in a general way, we plotted all data points of the joint tuning function against the prediction of an additive combination of single feature tuning curves and fitted the data with the exponential function $y_{\text{joint}} = m (x_{\text{add}} - b)^c + o$. An exponent of 1 indicates an additive mode of integration, 2 a multiplicative one. **Supplementary Fig. 7a** shows four examples of such scatter plots together with their fits, and **Supplementary Fig. 7b** shows the distribution of exponents of all fits. Exponents ranged from 0.7 to 3.9. About three quarters (78%) of exponential fits were compatible with additive, multiplicative, or some intermediate form of feature integration (exponents between 0.9 and 2.1). Twelve percent of all fits were sub-additive (exponent < 0.9), and ten percent supra-multiplicative (exponent > 2.1).



Supplementary Figure 7 Singular value decomposition analysis and polynomial fitting of joint tuning functions. **(a)** Four examples of polynomial fits to joint tuning functions. **(b)** Distribution of exponents of all fits ($n = 771$).

Thus middle face patch neurons use a range of integration schemes from the sub-additive to the supra-multiplicative, providing a rich repertoire of feature integration in face space.

Reference List

1. A. Grunewald and E. K. Skoumbourdis, "The integration of multiple stimulus features by V1 neurons," *J. Neurosci.* 24(41), 9185 (2004).
2. J. A. Mazer, *et al.*, "Spatial frequency and orientation tuning dynamics in area V1," *Proc. Natl. Acad. Sci. USA* 99, 1645 (2002).
3. J. L. Peña and M. Konishi, "Auditory spatial receptive fields created by multiplication," 292, 249 (2001).

Supplementary Text 5: Finding significantly tuned dimensions using Gaussian fitting

We also used fitting of a Gaussian function to find significantly tuned dimensions. We fitted a generalized Gaussian with four free parameters of the form

$$y = o + a \cdot \exp(-((x-p)/(w/2))^2)$$

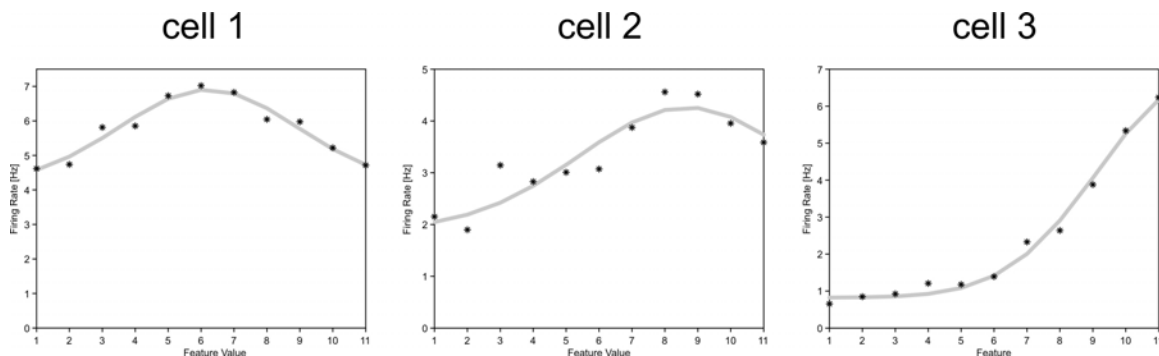
to each tuning curve (o offset, a amplitude, p peak position, and w peak width). As fitting algorithm we used the trust-region method based on the interior-reflective Newton method (implemented in the MATLAB routine `lsqcurvefit`). We resorted to this algorithm, after trying a number of different ones, because it allowed for the specification of parameter borders. Specifying these borders turned out to be critical for successful fits of ramp shape tuning curves. The reason is that when these curves are to be described by Gaussians, this can be done by a virtually infinite number of parameter combinations with almost identical shape over the relevant region of independent values (feature values 1 to 11). This is because only one Gaussian flank is fitted to the actual data, while center position, amplitude and offset are largely unspecified. Since most of our tuning curves were ramp-shaped, it was critical to address this problem by limiting the range of possible parameters.

Goodness of fit was assessed by the χ^2 metric¹. Following Young et al.², we required the χ^2 value to be at least 15% smaller than the variance of the data. We then tested for significance of the amplitude parameter at $p=0.01$. This level was chosen to limit the false detection rate. Thus, as is customary (e.g., Press et al., 1986), we required successful convergence of the fit, a sizeable reduction of unexplained variance and significance of the one (out of four) parameters that is most important for description of tuning. We tested the fitting routines with two large test data sets of several thousand artificial tuning curves. The first set consisted of artificially generated Gaussian shaped tuning curves

with amplitudes at least 1% of the height of the offset. More than 97% of the Gaussian tuning curves were found by Gaussian fitting. The second data set consisted of 11-element random data vectors. Using the random data, we determined the false positive rate of the algorithm to be only slightly higher than the expected 1%.

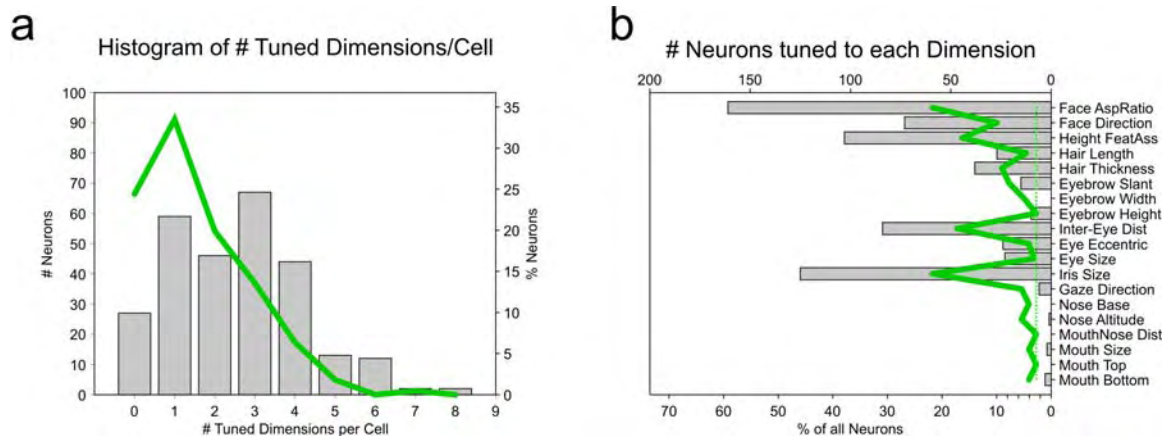
We then applied this method to the actual data. Three example fits are shown in

Supplementary Fig. 10.



Supplementary Figure 10 *Tuning in three example cells assessed by Gaussian fitting and Entropy-based methods. Tuning was significant by both methods.*

We applied the Gaussian fit method to our entire dataset and compared results obtained with our data reshuffling entropy based method (referred to from now on as the Gaussian and the Entropy method, respectively). The Gaussian method found fewer significantly tuned dimensions than the Entropy method. With the Entropy method, we had found 695 dimensions to be significantly tuned, rendering 90.1% of all 272 cells to be significantly tuned. With the Gaussian method 412 dimensions were found to be significantly tuned, rendering 75.6% of the cells to be significantly tuned. Those cells that were found to be tuned, were, on average tuned to fewer dimensions (2.0 vs. 2.8 for the Entropy method), as is detailed in **Supplementary Fig. 11a**. The Gaussian method found significant tuning in all 19 stimulus dimensions with preference for the eye and facial layout parameters as the Entropy method, but this preference was less pronounced than it was for the dimensions found with the Entropy method (**Supplementary Fig. 11b**). It is noteworthy that face aspect ratio is the feature dimension with the relatively highest incidence of non-



ramp-shaped tuning curves and the dimension on which the Gaussian method

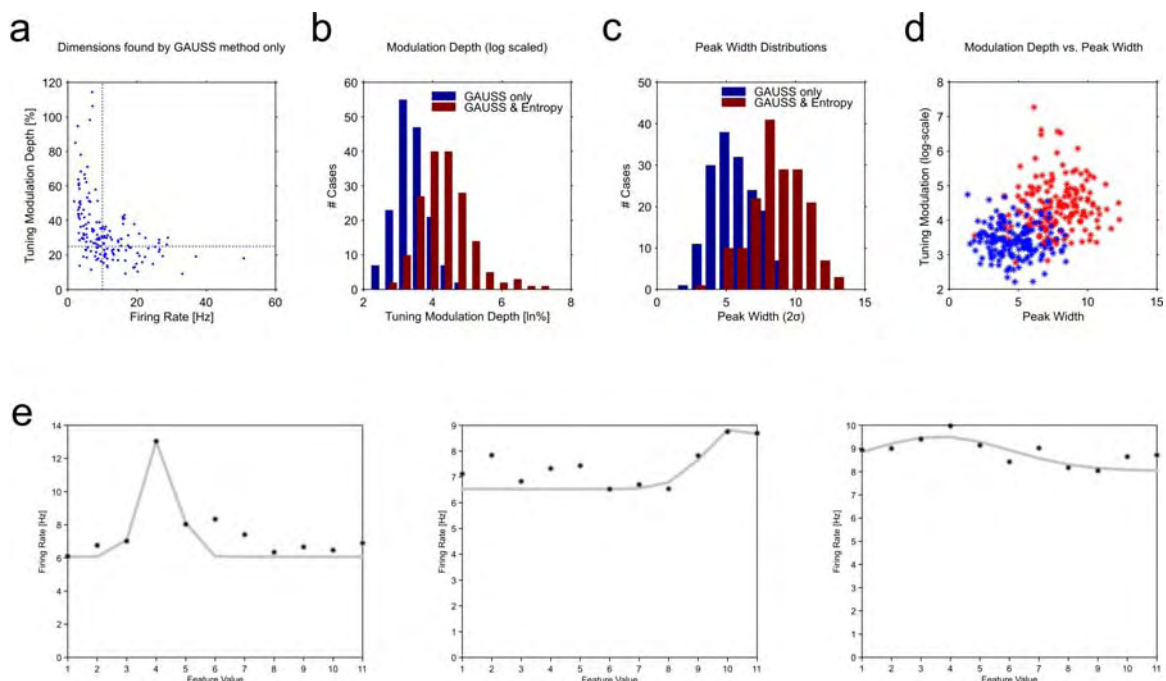
Supplementary Figure 11 Incidences of tuning as assessed by the Gaussian (green line) and Entropy based methods (gray bars). **(a)** Number of neurons with significant tuning to n dimensions (cf. manuscript Fig. 3c). **(b)** Number of neurons tuned to each dimension (cf. manuscript Fig. 3d), dotted green line indicates expected false detection level of the Gaussian method (1%, as validated in randomly generated

“underperformed” the most, indicating the drop in significant tuning curves found by the Gaussian method is not simply due to fitting problems with ramp-shaped tuning curves.

When we asked which fraction of the tuning curves found to be significantly tuned by the Entropy method, were also found to be significantly tuned by the Gaussian method. The number was 52%. Thus, the Gaussian method misses almost half of the dimensions found to be significant by the Entropy method. In hindsight, this may not be that surprising for several reasons. First, the Gaussian method assumes a certain shape of tuning – major deviations from that shape are not considered. This was likely not the decisive factor though, since the majority of tuning curves found with the Entropy method could well be approximated by a Gaussian. More importantly, however, the power of the Gaussian method is limited, because it is only based on 11 data points, the average tuning curve, to fit a function with four parameters. (We tried several ways to overcome this limitation, especially by splitting the data and doing multiple fits or by fitting several dimensions at once, but none of these methods proved to be robust.) In contrast, the Entropy method directly takes into account the full wealth of a data set obtained by several thousand stimulus presentations, because the entropy value is tested against the entire distribution

of shift predictors, and thus tuning can be detected with higher statistical power. In brief, fitting to the average tuning curve cannot take into account the variability of the data. Therefore, the power of the method to differentiate between true tuning peaks and randomly occurring ones (and spikes occur frequently in randomly generated data, especially with cells of low firing rates) is low.

The finding that only 52% of the dimensions the Gaussian method deemed significantly tuned had been found by the Entropy method, implies that the Gaussian method did find a sizable number of dimensions which the Entropy method had not found. We next investigated the properties of these “new” tuning curves. Do they point out a limitation of the Entropy method? First, it should be remembered that when we employed the Entropy method, for a dimension to be called significantly tuned, we had applied two criteria that we did not use in the case of the Gaussian method (because it would then have found



Supplementary Figure 12 Results of Gaussian fitting method. (a) Scatter plot of modulation depth and firing rate of all tuning curves the Gaussian method, but not the Entropy method had labeled significant. Dotted lines mark 10Hz firing rate and 25% modulation depth, respectively. (b) Comparison of modulation depth for Gaussian fits that the Entropy method found to be significant, too (red), or the Entropy method found not to be significant (blue). (c) Same as b, but for tuning curve width. (d) Scatter plot, of all significant Gaussian fits, modulation depth vs. peak width. (e) Three examples of tuning curves the Gaussian, but not the Entropy method found to be significant. Case on the left shows narrow peak. Maximal response value may or may not be an outlier response.

even fewer dimensions). We had required the dimension to be significantly tuned at two subsequent time intervals (spaced at least two times the temporal smoothing kernel width apart from each other). We had further required a modulation depth of the tuning curve of 25% or more. As it turned out, 36% of the dimensions found significantly tuned by the Gaussian method, would not have met that minimal tuning modulation depth criterion (**Supplementary Fig. 12a**). The second striking feature of this plot is that the majority of tuning (55%) is found for low firing rates (less than 10Hz). Second, we inspected the shape of the tuning curves more closely and compared tuning that both Gaussian and Entropy method had found significant (**Supplementary Fig. 11**) with those that only the Gaussian method had found significant (**Supplementary Fig. 12e**). The modulation depth was significantly higher for the former group of tuning curves than for the latter (average tuning depth 111% vs. 34%, $p < 0.001$, Mann-Whitney U-test, **Supplementary Fig. 12b**). Furthermore tuning width was on average narrower for the tuning curves only the Gaussian method had detected (2σ : 4.7 vs. 7.7 feature values, $p < 0.001$, Mann-Whitney U-test, **Supplementary Fig. 12c**). Thus in a scatter plot of tuning curves along these two parameters (**Supplementary Fig. 12d**), these two sets of tuning curves are clearly distinct.

More weakly and more narrowly tuned tuning curves are, as our tests with randomly generated data showed, more susceptible to be confused with non-systematic variations of tuning curves, especially in cases of low firing rates. In these cases the Entropy method would be more conservative, because the set of shift-predictors would contain similarly “spiky” curves, because the shuffle predictors would be similarly noisy as the tuning curve. In other words: tuning curves were deemed insignificant by the Entropy method, because (a) their tuning was weak, (b) because tuning was not reproducible at temporal separations of ≥ 30 ms, or (c) because similar tuning modulations were generated from shift predictor data. Note that the expected number of false positive

tuning dimensions of the Gaussian fit method was 52 for our data set (at a $p=0.01$ significance level). Thus, the Gaussian fitting method does not give us much confidence to securely declare an observed tuning to be real.

In summary, we found the Gaussian fitting method, though more prone to a higher false positive detection rate, to be less powerful and reliable in finding significantly tuned dimensions. Therefore, in light of these methodological results, we think that this method does not present an attractive alternative the reshuffling method we have used in our manuscript. We therefore think that the result of Gaussian fitting supports our decision to use the Entropy method instead.

Reference List

1. Press, W.H., Flannery, B.P., Teukolsky, S.A. & Vetterling, W.T. Numerical Recipes in C. Cambridge University Press, Cambridge, New York, Port Chester, Melbourne, Sydney (1988).
2. Young, M.P., Tanaka, K. & Yamane, S. On oscillating neuronal responses in the visual system of the monkey. *J. Neurophysiol.* **67**(6), 1464-1474 (1992).